Software Engineering Institute

Standards-Based Automated Remediation: A Remediation Manager Reference Implementation, 2011 Update

Prepared for the
National Security Agency Security Automation Office

Sagar Chaki, Software Engineering Institute
Rita Creel, Software Engineering Institute
Jeff Davenport, Software Engineering Institute
Mike Kinney, National Security Agency
Benjamin McCormick, Software Engineering Institute
Mary Popeck, Software Engineering Institute

December 2011

SPECIAL REPORT
CMU/SEI-2011-SR-016

Acquisition Support Program; CERT® Program; and Research, Technology, and System Solutions Program

http://www.sei.cmu.edu

Carnegie Mellon

This material is based upon work funded and supported by the United States Department of Defense under Contract No. FA8721-05-C-0003 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center.

This report was prepared for the

SEI Administrative Agent
ESC/CAA
20 Schilling Circle, Building 1305, 3rd Floor
Hanscom AFB, MA 01731-2125

Table of Contents

List of Figures

List of Tables

Acknowledgments

The authors would like to acknowledge the following members of the Software Engineering Institute's CERT® Program, who contributed to the project by providing engineering and domain knowledge and support and participating in meetings and teleconferences: Rex Brinker, Chad Dougherty, Allen Householder, Chris Inacio, Drew Kompanek, Marty Lindner, and Art Manion. We also appreciate Tamara English's administrative support and Paul Ruggiero's editorial support. We thank Joe Wolfkiel of the Defense Information Systems Agency (DISA) for his guidance early in the project and for his continuing interest in our work. Finally, we extend appreciation to MITRE collaborators Matthew "Woj" Wojcik and Gerry McGuire and SPAWAR Systems Center Atlantic collaborators Jack Vander Pol, Kyle Stone, and Richard Kelly. Successful completion of this work would not have been possible without their contributions.

Abstract

This report describes the Software Engineering Institute's (SEI's) 2011 work for the National Security Agency (NSA) to develop standards for automated remediation of vulnerabilities and compliance issues on Department of Defense (DoD) networked systems. The SEI developed a remediation manager reference implementation that demonstrates how evolving standards can communicate and process information on vulnerabilities, compliance issues, remediation policy, and remediation actions. An earlier report, *Standards-Based Automated Remediation: A Remediation Manager Reference Implementation* (CMU/SEI-11-SR-007), described the project's concept, vision, scope, requirements, and the remediation manager implementation as of December 30, 2010. Since then, the SEI has analyzed additional user scenarios, continued remediation standards development, and added new capabilities to the reference implementation.

The remediation manager can employ standards throughout the compliance issue remediation cycle. Using common formats and languages, the reference implementation ingests scan findings, extracts host compliance issues and vulnerabilities, maps them to remediation actions, builds remediation tasks, transmits remediation tasks to a Remediation Tool on a host system, and receives remediation task execution status from the Remediation Tool. In 2011 the SEI added a standards-based remediation policy management capability, enabling users to examine, tailor, and apply standard DoD policy to meet local needs.

1 Introduction

In 2010 the Software Engineering Institute (SEI) of Carnegie Mellon University began a research project, sponsored by the National Security Administration, on the remediation of computer vulnerabilities and compliance issues. The key goal of this project was to develop and test standards that support the automated remediation of vulnerability and compliance issues on DoD networked systems.

This report describes the project in general and provides an overview of the SEI's role, which has been to develop a reference implementation for a remediation manager. This effort has been central to advancing emerging remediation automation standards. In developing the reference implementation, the SEI has elicited and elaborated user scenarios and requirements for both the standards and an operational remediation manager, developed software for a variety of remediation management functions that demonstrate the use of the standards, and identified and documented considerations and candidate next steps for the way ahead.

This report is the second on this project. The first, *Standards-Based Automated Remediation: A Remediation Manager Reference Implementation* (CMU/SEI-11-SR-007), documented our work in calendar year 2010. The current report recaps the purpose, vision, and scope of the effort and describes the work we have accomplished in calendar year 2011.

1.1 Remediation Research Overview

Existing methods and tools for remediating vulnerabilities and misconfigurations of Department of Defense (DoD) networked systems either rely heavily on manual support, which is inefficient and error prone and complicates delivery of remediation status data, or rely on proprietary vendor solutions.[1] The Remediation Research Project seeks to address these problems by (1) developing remediation standards, (2) increasing the efficiency and effectiveness of remediation by automating a remediation process that ensures host configurations comply with DoD policy, and (3) standardizing remediation processing.

The Remediation Research Project consists of four elements of work that advance efforts to develop standards-based, automated remediation capabilities:

- *remediation automation standards* (MITRE, NIST, Software Engineering Institute [SEI], SPAWAR Systems Center Atlantic, National Security Agency [NSA])
- *sample content*—security-related checklists, enumerations, and other information created in accordance with existing Security Content Automation Profile (SCAP) standards as well as the emerging remediation automation standards we are working to develop and test (G2, MITRE, NSA, SPAWAR Systems Center Atlantic)

[1] A *vulnerability* is a state in a system that allows an attacker to execute unauthorized commands, bypass restrictions on data access or modification, pose as another entity, or affect the availability of a system resource. A *misconfiguration* is any configuration state that does not comply with an organization's security policy. A *remediation* is a security-related set of actions that result in a change to a computer's configuration that brings it into compliance with policy (e.g., to address a vulnerability or misconfiguration) [Waltermire 2011, p. 1].

- a *Remediation Manager reference implementation*—the subject of this special report (SEI)
- a *SCAP-based compliance checker* and *Remediation Tool reference implementation* (SPAWAR Systems Center Atlantic)

The remediation automation standards component of this work is based on the Derived Requirements (DR) identified by Waltermire, Johnson, Kerr, Wojcik, and Wunder [Waltermire 2011], which are shown in Table 1.

Table 1: Derived Requirements for Remediation Standards [Waltermire 2011]

ID #	Derived Requirement / *Abbreviation for Standard Item*
DR1	method for uniquely identifying a remediation / ***CRE***
DR2	definition of an exchange format for basic remediation information / ***CRE-DEF***
DR3	definition of desired additional data about a remediation, including mappings to applicable platforms, related vulnerabilities, or configuration issues / ***ERI***
DR4	definition of an expression language for the additional data about remediations as identified in DR3 / ***ERI-DEF***
DR5	method for specifying which remediations apply to which classes of assets / ***RPL***
DR6	method for applying specific remediations to specific assets in an enterprise environment / ***RTL***
DR7	method for reporting the results of an attempted remediation / ***RRF***
DR8	method for expressing how to perform a remediation in a precise, machine-readable fashion *[Note: DR 8 is not part of the work described in this report and was rejected as a pursuit due to projected cost, complexity, concerns regarding the likelihood of success, and lack of vendor support.]*

CRE	Common Remediation Enumeration
DEF	Definition
ERI	Extended Remediation Information
RPL	Remediation Policy Language
RRF	Remediation Results Format

Sample content, for use by the reference implementations, has been created as work on the standards progresses. Both the remediation standards and sample content are works in progress and should not be considered final.

The Remediation Manager (RM) reference implementation[2] ingests scan results, in DoD Assessment Results Format (ARF) version 0.41, which contain findings from host scans in the form of Common Configuration Enumeration (CCE) and Common Vulnerabilities and Exposures (CVE) entries. The Remediation Manager reads the policy for the given host's policy group. This policy maps CVEs and CCEs to a corresponding Common Remediation Enumeration (CRE) entry. Using the CRE, the Remediation Manager builds remediation tasks and transmits these tasks in Remediation Tasking Language (RTL) to the Remediation Tool (RT) associated with the host machine that requires remediation.[3] The Remediation Tool returns results to the Remediation Manager in Remediation Results Format (RRF). The Remediation Manager maintains a log indicating remediation task status (in process, failed, accomplished, not applicable, or undefined).

Figure 1 illustrates the role of the emerging standards in automated remediation management.

---

[2] The purpose of the reference implementation is to support development of remediation standards. The 2010 version does not incorporate all essential capabilities and quality attributes and is not a basis for operational system development.

[3] The 2010 Remediation Manager reference implementation accommodates scan results in DoD ARF version 0.41. Future versions will also accommodate scan results in Assessment Summary Results (ASR), eXtensible Configuration Checklist Description Format (XCCDF), and Open Vulnerability and Assessment Language (OVAL).

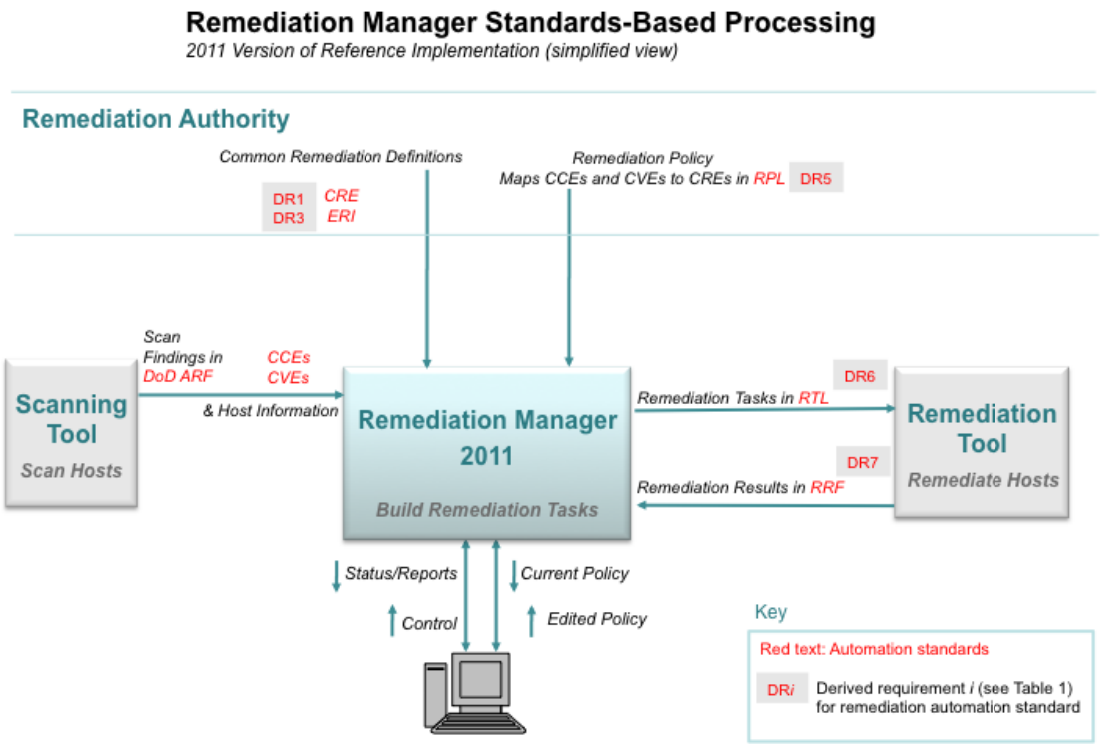


Figure 1: Standards-Based Processing for Automated Remediation Management, 2011 Update

The Remediation Tool reference implementation consists of software that resides on a host system. This software receives remediation tasks from the Remediation Manager, executes these tasks on the host system, and sends task execution status back to the Remediation Manager. Initially, the Remediation Tool will be limited to remediating registry keys, file permissions, and local policy changes.

Note that in the 2010 implementation, the Remediation Manager assumes that the mapping from each scan finding (CCE or CVE) to each CRE has been defined in DoD remediation policy, and there is no need for user intervention. Future systems will include the ability for users to select from multiple CREs when necessary, to override DoD policy with local policy for machines that belong to certain policy groups, to choose a mitigation action, or not apply a remediation or mitigation at all. The user will also have the capability to enter justifications and build a Plan of Action and Milestones (POA&M) to handle deviations from DoD policy and output the results using SCAP version 1.2 standards.

1.2 Purpose and Organization of Report

The purpose of this report is to document the work accomplished on the Remediation Manager reference implementation in 2011, and to provide a technical foundation—including user scenarios, requirements, top-level architecture, and capabilities descriptions—for future work. In addition to describing the 2011 implementation, the report includes information on a broader set of requirements and on questions to consider in defining the way ahead.

The report is organized into the following sections:

1. Introduction

2. Vision, Scope, and Approach
3. User Scenarios, Remediation Standards, and Requirements
4. Current Reference Implementation Architecture
5. Current Reference Implementation Capabilities
6. Observations, Next Steps, and Conclusions

Sections 1 and 2 of this report are similar to the corresponding sections in our previous report, with updates to figures and reference documents that reflect work completed in 2011. Section 3 expands and revises our 2010 user scenario and requirements work. In 2011, we used a wiki to document and exchange ideas on user scenarios and requirements. The material in Section 3 captures the state of the scenarios and requirements on the wiki as of September 30, 2011. Section 4 provides a brief overview of the reference implementation architecture, tracing the top-level remediation manager capabilities and requirements to the top-level functions and architectural components of the 2010 and 2011 reference implementations. It also discusses the relationship of SCAP and emerging remediation automation standards to the architecture.

Section 5 covers remediation manager capabilities. This section was not included in the 2010 report because the 2010 implementation centered on under-the-hood functionality rather than capabilities visible to the user. Section 5 provides screenshots from the 2011 remediation manager reference implementation with a description of the capabilities provided on each screen. Section 6 identifies questions for consideration and candidate next steps for standards-based, automated remediation work.

2 Vision, Scope, and Approach

2.1 Overview

This section describes the vision for the desired remediation management solution. It presents ideas developed during NSA's envisioning phase, describes the current context and the vision for the future, identifies key remediation management features, and illustrates the conceptual solution structure. Some of the goals for the Remediation Manager are implemented in the 2010 reference implementation, and others will be achieved through continued development in 2011.

2.2 Remediation Management Context

The Remediation Manager development effort has been defined to fit within the notional DoD network configuration management hierarchy shown in Figure 2, which illustrates the objective of leveraging standard remediations and policies defined at the highest level. While the objective of such reuse is laudable, lower-level tiers must retain the ability to tailor remediations and policies to address their respective mission objectives and risks.

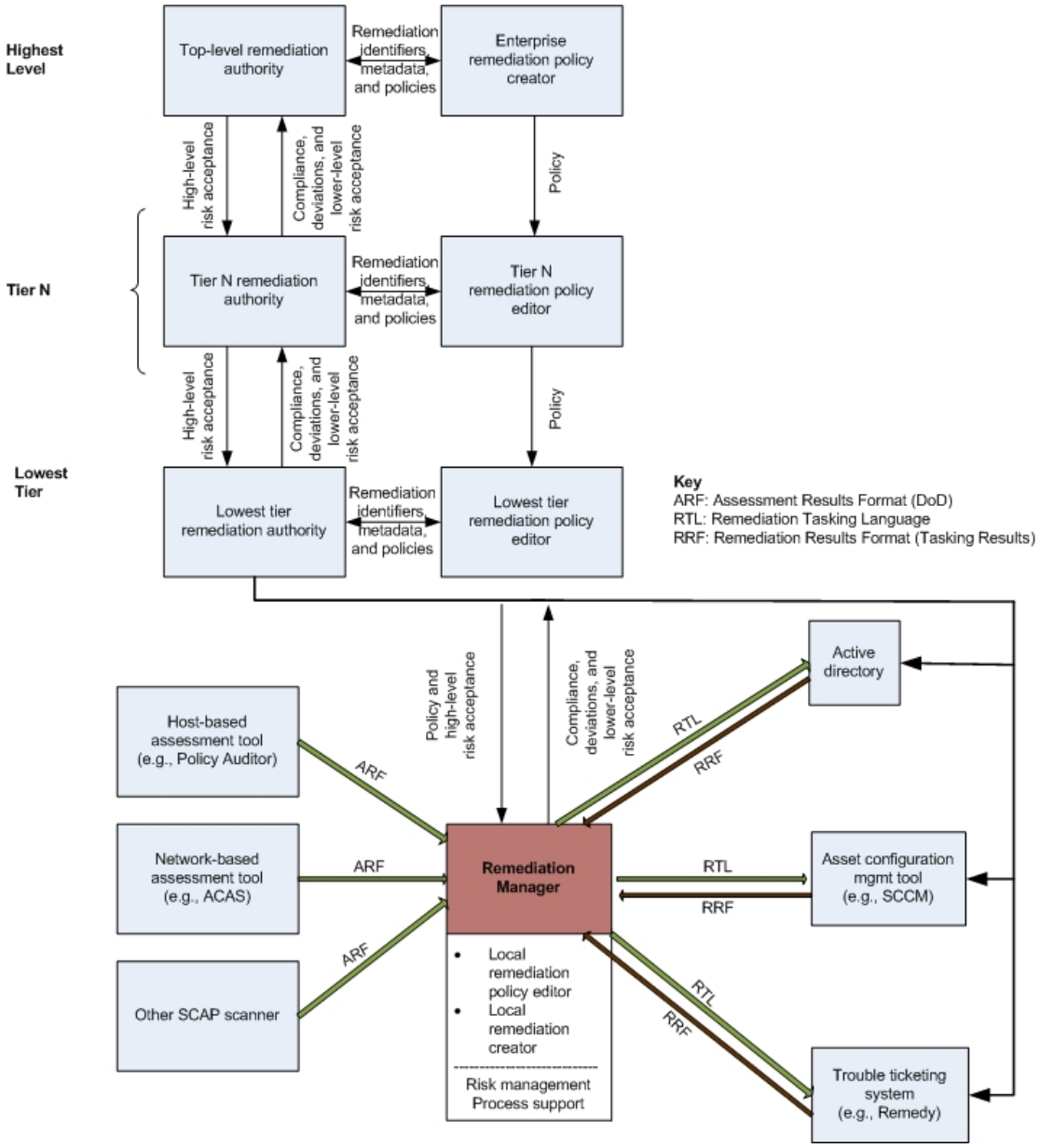


Figure 2: DoD Configuration Management Process Vision, Adapted from DoD[4]

At the bottom tier shown in Figure 2, assessment tools scan hosts (perform assessments) and produce results in standard formats. The use of standard formats for assessment data—which can be loaded into a repository, aggregated, correlated, deconflicted, interpreted, and processed—enables the following capabilities:

- Users of the assessment tools can manually organize, visualize, and understand assessment data.
- The Remediation Manager described in this document can ingest the assessment data, along with remediation policy instructions that map assessment findings to remediation directives, and automatically output a directive (task) to apply a remediation.

[4] U.S. Department of Defense. *Operational Concept Summary*. DoD, undated.

The focus of this document is on the capabilities, characteristics, and development of a reference implementation for the Remediation Manager. The Remediation Manager will implement automated delivery and execution of remediation directives (also called tasks) for systems on DoD networks.

The work accomplished in developing the Remediation Manager reference implementation is expected to facilitate the DoD's procurement of standards-based remediation solutions via

- vendor development of standards-based, off-the-shelf components for various elements of the remediation solution
- an acquisition approach for an operational Remediation Manager implementation that is evolvable in capability and scale and meets specified functional, performance, and quality attribute (i.e., supportability and dependability) requirements

2.3 Significance of Standards-Based Automated Remediation

The DoD relies heavily on networked assets to perform its missions. These assets, and their interconnections, continue to grow in number and complexity. Maintaining a secure configuration—ensuring critical patches, settings, and updates are applied—is an ongoing challenge. The SCAP suite of security standards provides a means to express information about the configuration of networked assets and the results of scans so that prompt remediation and mitigation actions can be implemented. Emerging remediation standards will likewise provide an approach to expressing, selecting, and applying remediations to assets that are out of compliance or vulnerable to attack.

For remediation and mitigation to be prompt, automation is essential. The goal is to implement a standards-based, automated remediation solution that can be deployed within the DoD on enterprise-wide or isolated network enclaves (e.g., a tactical environment) to ensure that vulnerabilities and issues of noncompliance with DoD policy and guidance are corrected as soon as possible. Note that the DoD is not the only U.S. government department to promote standards-based security automation. In a recent white paper, the Department of Homeland Security (DHS) identified interoperability components (such as SCAP standards and content) and automation as two of three building blocks of a healthy cyber ecosystem for the nation's critical infrastructure systems (the third building block is authentication) [DHS 2011].

The vision for advancing vulnerability and configuration policy compliance will be realized in an evolutionary fashion and is described in Table 2 in terms of

- the current process (manual, supported by scripting and local methods and tools)
- basic capabilities to be provided by the reference implementation (2010 and 2011)
- capabilities under consideration for an initial operational capability to be acquired
- the desired final operational vision

Table 2 represents the current, top-level understanding of desired capabilities. This understanding will evolve as work to develop remediation standards continues.

Table 2: Remediation Manager Evolutionary Vision

Remediation Process Requirements (Current Process)	**Current Process: Manual Approach to Remediation** • Local information assurance (IA) users perform compliance scans and identify items to be remediated. • Scan results provided to local administrators who remediate manually, using scripting, local tools, and other methods. • IA users rescan and obtain Designated Approval Authority (DAA) acceptance for discrepancies. • IA users report results up the chain of command.
Refined Requirements (Reference Implementation)	**Reference Implementation: Research and Development of Automated, Standards-Based Remediation Manager** • Demonstrate how current scanning and remediation processes can be integrated and automated using SCAP and emerging remediation standards. • Perform automated compliance remediation actions based on preapproval of remediations (2010). • Automate remediation reporting, include POA&M(s) and statements of risk (2011).
Refined Requirements & Architecture (Initial Operational Implementation)	**Initial Operational Remediation Manager: Current Concept** • **Scope:** Limited scanning tool input and standards-based remediation policy govern limited patch and software setting configuration modifications. • **Remediation Policy:** The Remediation Manager (RM) ingests and stores CREs, ERI, (information associated with CREs), and remediation policy XML, which represents the baseline policy for all policy groups. Remediation policy maps CREs and required parameters to CVEs and CCEs. Using the local policy editor function of the RM, the local RM administrator may elect to apply local remediation policy rather than higher-level policy to a policy group of hosts. Overriding higher-level policy requires a justification/POA&M to document accepted risk and a time line (deadline) to bring the asset into compliance, which is reported in the remediation results. • **Host Assignment to Policy Group:** An administrator assigns each host in the Remediation Manager's inventory to the policy group that determines the set of remediation policies applied. The administrator may later decide to move a host into a different policy group. • **Scanning:** Scan results are sent to the Remediation Manager in the form of an Assessment Results Format (ARF) XML document. The Remediation Manager ingests the scan results and extracts findings (CCEs and CVEs). • **Policy Assignment for New Findings in a Policy Group:** The first time a finding is encountered for a host in a given policy group, the administrator is prompted to select default higher-level policy or local policy for that finding, for hosts in that policy group. If the finding requires different treatment for some hosts in a policy group, the administrator can move these hosts to a different policy group. • **Remediation for New Findings on a Host:** Once the remediation policy for a finding has been verified for a policy group, when a host in that group first encounters the finding, a remediation task will be sent to the appropriate Remediation Tool for execution on that host. The task status will be marked "in process" until the Remediation Tool returns a result status to the Remediation Manager. If remediation status is "failed," the machine is flagged and a ticket created so the host can be manually checked and remediated. • **Remediation for Repeat Findings on a Host:** If the host was previously scanned and tasked for remediation and the same finding is identified after the remediation deadline, the host is flagged and a ticket created requiring the host to be manually checked and remediated. • **Reporting:** Periodically, a report will be generated on all hosts in the Remediation Manager's inventory indicating findings from scan results and remediation status. This report will be an ASR report supplemented as needed to show remediation status information. The Remediation Manager administrator or user can manually generate a report at any time and display it at the Remediation Manager console. • **Remediation Manager Requests for Scans:** Newly discovered hosts will be placed into an "unassigned" group pending assignment to a policy group by the Remediation Manager administrator. When a new host is placed in a policy group, the Remediation Manager will prompt the administrator to request compliance scans or to request automated remediation without requiring a scan.
Refined Requirements & Architecture (Vision)	**Vision** • **Scope:** All devices on a network. • **Capability:** Data and logic to determine the best remediation option for a given host.

2.4 Vision Statement for Remediation Manager Reference Implementation

The Remediation Manager reference implementation should achieve the following objectives:

- Demonstrate the features documented in Section 2.5 of this document. (Remediation Manager increments developed in 2010 and 2011 demonstrate a subset of these features.)
- Support development of standards and associated content.
- Interface with others who are working on various aspects of security automation.
- Enhance understanding of the desired features of an operational Remediation Manager and remediation product suite implementation.
- Provide a technical foundation for development, procurement, or acquisition of an Operational Implementation via insights gained through reference implementation activities.

2.5 Remediation Manager System-Level Functional Requirements

Table 3 lists the top-level functional (feature) requirements defined for the Remediation Manager. Section 3.3 decomposes these requirements and allocates them to Remediation Manager components. It also identifies nonfunctional (quality) requirements. Note that not all system-level requirements have been implemented in the 2010 and 2011 versions of the reference implementation, and some system-level requirements are only partially implemented. In some cases, this is because the necessary standards and content are not yet available; in others, it is because the functions were not allocated for implementation in the reference implementation but were left for vendors to develop in an operational system.

Table 3: System-Level Functional Requirements

#	System-Level Requirement	Appendix A Reference
1	Accept input scan results formatted as • DoD's ARF version 0.41 (implemented in 2010) • Assessment Summary Results (ASR), XCCDF, and OVAL (possible future)	Sys Remediation Manager 2.1 (ARF)
2	Accept input policy instructions consistent with standards-Derived Requirement DR5 [Waltermire 2011, NSA 2010[5]] (future).	Sys Remediation Manager 2.2
3	Output a directive to apply a remediation per standards-Derived Requirement DR6 [Waltermire 2011, NSA 2010[6]] (implemented in 2010).	Sys Remediation Manager 3.1
4	Allow users to choose which remediation to apply when multiple options are included in the policy (future).	Sys Remediation Manager 4.2
5	Determine the most efficient method of remediation (e.g., applying a single patch to fix multiple vulnerabilities) (possible future).	Not Applicable
6	Decide how to remediate when multiple remediation systems, including network-oriented systems, are available (possible future).	Not Applicable
7	Allow a user to tailor remediation policy for a given set of assets as well as accept some risks (i.e., decide not to remediate) (future).	Sys Remediation Manager 4.3
8	Assist users in building POA&Ms for policy deviations (future).	Sys Remediation Manager 4.4
9	Provide capability to publish POA&M messages consistent with Netops data standards (future). Note: For the reference implementation, when a deviation from policy is detected, the expected level of capability will be to reference a POA&M or make a mitigation statement (i.e., full POA&M capabilities are not a high priority for the reference implementation).	Sys Remediation Manager 3.5
10	Accept Remediation Tool results per standards-Derived Requirement DR7 [Waltermire 2011, NSA 2010[7]] (implemented in 2010).	Sys Remediation Manager 2.5
11	Republish findings received from the Remediation Tool with notations on fixes made (e.g., updating XCCDF results type to "fixed," adding "info" messages to OVAL) (future).	Sys Remediation Manager 3.2

2.6 Remediation Manager Top-Level Functions

The Remediation Manager requirements identified in Table 3 above can be grouped into four main functions:

1. Stage and Manage Policies (requirements 2, 4, 7, 8, and 9)
2. Ingest Findings and Build Policy-Based Remediation Tasks (requirements 1, 5, and 6,)
3. Transmit Remediation Tasks to a Remediation Tool and Accept Results (requirements 3 and 10)
4. Report Remediation and Risk Status (requirements 9 and 11)

The *Stage and Manage Policies* function obtains remediation policies, CREs, and Extended Remediation Information (ERI) from a remediation authority and prepares them for use by the Remediation Manager. This preparation may include tailoring of DoD-level policy to meet local requirements. A justification must be written for any tailoring of policy; the intent is to implement

[5] U.S. National Security Agency. *Integrated Statement of Work for FY2010 Remediation Concept Development.* NSA, 2010.

[6] U.S. National Security Agency. *Integrated Statement of Work for FY2010 Remediation Concept Development.* NSA, 2010.

[7] U.S. National Security Agency. *Integrated Statement of Work for FY2010 Remediation Concept Development.* NSA, 2010.

a POA&M capability for this purpose, as part of the *Stage and Manage Policies* function and *Report Remediation and Risk Status* function (see requirement 9 in Table 3).

The *Ingest Findings and Build Policy-Based Remediation Tasks* function extracts CCEs, CVEs, and host information from scans in DoD ARF version 4.1. It creates remediation tasks for these CCEs and CVEs based on the staged policies, which map each CCE or CVE to a CRE. Remediation tasks are compiled into an RTL file.

The *Transmit Remediation Tasks to a Remediation Tool and Accept Results* function transmits remediation tasks (in RTL) to Remediation Tool(s) or other mechanisms that will accomplish remediation, and receives remediation results (in RRF) from these tools or mechanisms.

The *Report Remediation and Risk Status* function generates reports on the status of remediations and resultant risks, including risks and POA&Ms derived from policy exceptions.

In 2010, reference implementation work focused on the *Transmit Remediation Tasks to a Remediation Tool and Accept Results* function, with some work on *Ingest Findings and Build Policy-Based Remediation Tasks* function. In 2011, we worked on the *Stage and Manage Policies* and *Ingest Findings and Build Policy-Based Remediation Tasks* functions. The next section briefly describes the approach to our 2011 development effort.

3 User Scenarios, Remediation Standards, and Requirements

User scenario development and requirements analysis are critical tasks for advancing the concept of automated, standards-based remediation. First, these tasks elicit and clarify the vision and scope for an *operational remediation manager* and provide a firm foundation for architecting and developing a solution. Second, they expose requirements for *remediation standards* and provide a means to validate, early on, that the standards contain the elements needed to (a) automate remediation, (b) manage the application of standardized remediation policy, (c) enable standard data analysis from the end-to-end remediation process, and (d) report remediation tasking results and policy compliance patterns. Finally, user scenarios and requirements provide a means to evaluate and compare the functionality and quality characteristics provided by different remediation management solutions.

In 2010, we developed a base set of requirements by

1. reviewing top-level requirements
2. working with core members of the remediation community to identify key user scenarios
3. elaborating and documenting the user scenarios
4. allocating capabilities called out in user scenarios to one or more top-level requirements
5. refining and decomposing top-level requirements to capture the capabilities called out in user scenarios
6. posting the user scenarios and requirements to our team Remediation Research wiki for review and comment

The user scenarios and requirements we documented as a result of this process are provided in the next two sections. We expect work to continue on these scenarios and requirements as the remediation standards mature and the effort to build or buy a standards-based, automated remediation solution continues.

3.1 Remediation Manager (RM) User Scenarios

The scenarios in this section were developed based on requirements elicitation discussions in July and August 2011.

3.1.1 User Scenario 1: Ingest Policies, CRE Info, Scan Results and Other SCAP Info Needed by RM

The user goes to the Policy Manager Page and imports the appropriate DISA Security Technical Information Guide(s) (STIG or STIGs) which provides the RM with Baseline/Master policy information. The user also may import CRE/ERI, Common Configuration Scoring System (CCSS), Host information, Open Checklist Interactive Language (OCIL) information, and any other SCAP data as needed. The Remediation Manager (RM) stores all of this information.

The user goes to the Policy Manager Page and creates a local policy for a particular Policy Group (PG) based on the STIG, CRE, and OCIL information. The user verifies the new policy per User Scenario 6, then the RM maps this new local policy to that particular PG. Only one local policy item of a type (e.g., OS policy type) may be mapped to a PG, which can have any number of hosts mapped to it. Hosts may belong to only one PG. [Note: For an operational implementation, vendors would be expected to address conflicts when multiple policies exist.] If the local policy exactly matches the Master Policy that was received, then no POA&M is necessary. However, if the local policy differs from the Master Policy, then the user fills out a POA&M for the local policy and references the DAA. The RM maps this POA&M to the local policy and saves it. The user may transmit POA&M data to an external entity in accordance with the appropriate standard.

Periodically, the RM receives authenticated scan results on hosts in its purview. These scan results have been authenticated per the Scan Results Language Standard and are based on compliance to the appropriate Master Policy. [Note: The RM Reference Implementation does not have a trusted connection to the scan results provider. However, it is expected that an operational implementation would have one. The RM Reference Implementation only accepts ARF, but an Operational Implementation may accept results in other Host Scan Languages such as ASR.] The RM extracts finding information (i.e., CCE or CVE issue) from the scan results, marks new non-compliant findings as “new,” and stores the results. [Note: The RM uses the result (CCE) to look up the remediation (CRE) in the policy; this mapping is expected to be in the STIG. The RM Reference Implementation does not have knowledge of the benchmark that was used in the scan. An operational implementation should consider the need to identify the benchmark against which a scan was run and ensure it is the correct benchmark for the policy.]

Received scan results may contain information on hosts unknown to the RM. When this occurs, the RM places newly discovered hosts into the unassigned PG pending assignment by a user. [Note: In an operational implementation, the RM may have an ability to automatically assign newly discovered hosts to a Host Group (HG) or PG based on host information (e.g., operating system) identified in the scan.] Information on what Master Policy was used to perform the compliance scan is also received by the RM. After this information is received, the RM sends an alert to notify the user that a new host has been placed into the unassigned PG and what Master Policy it is expected to be in compliance with. This alert may be an email notification or other type of message and will be visible on the main page of the RM. The user may now assign the host to the appropriate PG either directly or through a Host Group (HG) assignment.

HGs are used to facilitate grouping and moving of large groups of similar hosts. An HG is mapped to only one PG. Because a host is permitted to belong to only one PG, it may only belong to one HG. A host does not have to belong to any HG; it may be placed directly into a PG. When using HGs, the user must assign the entire HG to a PG. Thus, placing a new host in an HG also places it in the appropriate PG. Multiple HGs may be mapped to one PG. Figure 3 illustrates this concept.

The local policy that is mapped to the PG may differ from the Master Policy used during the received scan. The RM will only generate tasks for those items that are not in compliance with the local policy mapped to the PG in accordance with User Scenario 2.

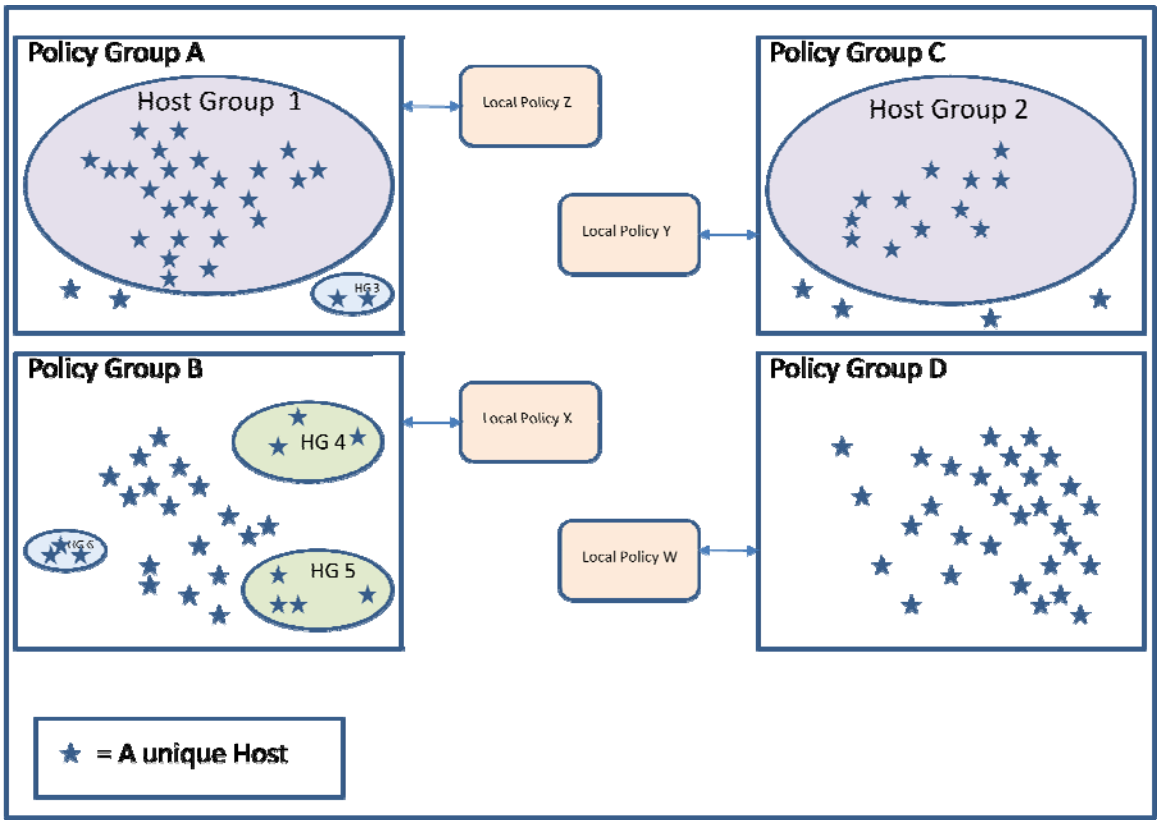


Figure 3: Host and Host Group Placement within Policy Groups

3.1.2 User Scenario 2: Create Remediation Tasks and Receive Status from RT

For hosts that are assigned to a PG and have scan results received, the RM determines if a particular host was previously scanned and if there were any open findings (i.e., CCE or CVE issues) against it. If the host was previously scanned and if the same finding is detected again in a new scan, then the RM determines if either the finding is against a policy item that is not contained in the local policy or if the remediation task that was generated to close the original finding has passed its due time. If the due time has not passed, then the RM just records the host's status. If the due time has passed, then the RM updates the remediation task status to "failed" and outputs a help desk ticket. If the finding is a repeated finding against a policy item not contained in the local policy, then the RM generates a help desk ticket. An alert is sent to the user either via email notification or other type of message that is also visible on the RM main page.

If a new finding is found against a host and the finding is against an item in the local policy, then the RM maps the new finding to its corresponding CRE(s) and generates a remediation task with a due time. The RM verifies that a specific CRE is only sent once to a specific host even if it has multiple CCEs that map to that CRE. The RM assigns the new remediation task to the appropriate Remediation Tool (RT) and transmits it to that RT per the specified RTL

standard. Once a remediation task is generated and sent, the RM sets the task status to "in process." If a new finding is against a policy item that is not contained within the local policy, the RM will just record this finding. Whenever this finding is received again, the RM will output a help desk ticket.

If no new findings are found against a host indicating that the host does not require remediation, then the RM simply records its status.

When the RM finds a noncompliant finding against a local policy item which has more than one potential remediation that could be performed to remediate the host, the RM generates a help desk ticket that is transmitted to the appropriate external entity in accordance with the help desk ticket standard. The RM also sends an alert to the user either via email notification or other type of message that is visible on the RM main page. This may be modified in an Operational Implementation to allow the user to select a remediation.

Results of a remediation task are returned to the RM from the RT per the specified Remediation Tasking Results Language standard. The RM assigns the following status to remediation tasks (note that the names for the states are unresolved and may change):

- "new" for tasks that have been generated but not yet sent to an RT
- "in process" for tasks that have been sent to an RT but no response has yet been received from the RT
- "completed" for tasks that the RT has responded to say it completed successfully
- "not completed" for tasks that the RT has responded that it has not yet completed but are not past their due date/time
- "failed" for tasks for which the RT has not sent any response and the due date/time has expired
- "reject" for tasks which the RT has responded that it will not complete or completed unsuccessfully
- "expired" for tasks which the RT has responded to say that it has not yet completed and the expected due date/time has passed

Table 4: Remediation Task Status Name and State

Status Name	RT ACK (y/n)	Task Success (y/n)	Task Expired (y/n)
new	-	-	-
in process - sent	n	-	-
in process - received	y	-	-
reject	y	n	-
failed/expired	y	-	y
not completed	y	-	n
completed	y	y	-

If the remediation task is tagged as “failed” or “expired,” then the RM creates a help desk ticket that describes the attempted remediation task and what failed. The RT shall have sent the information on what failed to the RM. The RM also sends an alert to the user either via email notification or other type of message that is visible on the RM main page. It is assumed that the RT either generates a POA&M and a help desk ticket for rejected tasks or provides the information to the RM so that the RM may generate the POA&M and help desk ticket. The details of when the RT creates POA&Ms and help desk tickets versus when the RM does still need to be resolved.

The Task Status Page shows all tasks generated by the RM and their status. A user views this page to determine if tasks have been generated and sent out to the RT and what responses have been received. Tasks are color coded based on their status listed above. The user may click on a specific task to bring up a page with all the information about that task. The user can tell if responses are past due and generate a help desk ticket if necessary.

The Host Management Page shows the status of all known hosts in the RM purview. Hosts are colored red if they have failed any remediation task. Hosts are colored yellow if remediation tasks that have been sent have not all been applied yet. Hosts are colored green if they are in compliance with their local policy and have no outstanding tasks or findings. Hosts are colored gray if there has been no scan received on them after a defined period of time. A user may click on a host to bring up a page with all the information about that host. The user then views the specific details of what remediation tasks have been sent to that host and what their current status is. Again, a user may decide to generate a help desk ticket if a particular remediation task was not successful or if a host has not been scanned in a long period of time.

3.1.3 User Scenario 3: Provide Remediation Reports

Periodically, the RM generates a remediation status report on all known hosts in its purview; this report is generated according to the specified standard and provides information on all

remediation tasks sent and their results and status along with the task due time. Report generation may be done via a commercial application, such as Crystal Reports, which interfaces with the RM. Reports may be automatically sent to recipients who have requested that service. In addition, the user may generate and display a manual report at any time. Users may generate reports on the content and hierarchy of policies as well as the health of all RM internal components. The user may also generate an event log of all RM actions over a given date range. The Report Manager Page provides multiple formats and methods for exporting reports. Reports may be generated to show trends, summaries, counts by remediation (e.g., CRE), and other indicators that indicate if systems are going out of compliance due to a Group Policy Object (GPO), newly applied patch, or some other reason.

3.1.4 User Scenario 4: Perform an Emergency Remediation

If a STIG or policy change of an urgent nature is received, the user goes to the Policy Manager Page and modifies the local policy that is mapped to the appropriate PG(s) with this new information. Before applying the policy change, the user tests the policy change on a Test PG per Scenario 6. If successful, the user requests that a special task(s) be generated by the RM to immediately send the request for all hosts in the target PG(s) to be brought in compliance with this new policy change. The RM does not wait for scan results in this scenario. The user then applies, and the RM maps, the newly modified policy to the target PG(s).

3.1.5 User Scenario 5: Move Host(s) to a New Policy Group

The user may move host(s) among PGs or HGs based on certain, and as yet to be defined, attributes and restrictions.

Hosts may be moved by the user from their current PG into a different PG or into an HG. Similarly, hosts may be moved by the user from their current HG into a PG or a different HG. They may be moved one at a time or in bundles by selecting as many hosts as desired on the Host Management Page. Similarly, a user may move an HG from its current PG into a different one.

Table 5, below, illustrates how hosts may be moved by the user.

Table 5: Host Assignments to Host and Policy Groups

From Current Location	To New Location
the “unassigned” PG	a PG
the "unassigned" PG	an HG
a PG	a different PG
an HG	a different HG
a PG	an HG
an HG	a PG

Hosts may be assigned to (or removed from) a PG or moved from one PG to another individually or in bundles by selecting as many hosts as desired on the Host Management page, or as an entire HG.

Hosts will not be sent tasks until scan results are received for them. The RM will only generate remediation tasks for findings against items in the local policy associated with the PG to which the host belongs. Other findings will be ignored the first time they are received; a help desk ticket will be generated thereafter.

3.1.6 User Scenario 6: Test a New Local Policy Before Sending Tasks to Hosts

Before a new policy is applied to the hosts in a PG, it is tested on a Test PG. The user places one Test Host into the Test PG and applies the new policy to this Test PG. The RM maps the new policy to the Test PG. The Test Host is known to be noncompliant with the new policy, and scan results that confirm this are imported. The RM then generates tasks to remediate the Test Host and sends these tasks to the appropriate RT. If the Test Host is remediated successfully, then the user applies this new policy to the target or intended PG. The RM discontinues mapping the new policy to the Test PG and now maps the new policy to the target PG. The next time scan results are received for hosts in that PG, tasks will be generated for findings that indicate noncompliance with the new local policy. If an emergency remediation is needed, then User Scenario 4 is performed.

3.1.7 User Scenario 7: Install and Set Up the Remediation Manager

An administrator installs the RM via the installation script and creates logins for all necessary roles (e.g., administrator, user). The administrator sets up optional items such as email notification, report generation, and report distribution list.

The administrator or a user then creates the various PGs that will be needed. The RM stores this information. The administrator or a user creates any HGs that will be needed. The RM stores this information.

If the RM is an embedded application, then it obtains its security settings from the host application.

3.2 Remediation Automation Standards

Table 6 lists the standards with which the RM is to be in compliance.

Table 6: Remediation Automation Standards [Waltermire 2011]

Std ID	Standard	Comments about what is needed in this standard
DR1	**Common Remediation Enumeration (CRE)** - Standard way of uniquely indentifying a remediation task	• Standard should express a definition for remediation tasks that includes parameter values in a predictable, parsable format. • Standard should include a mapping from CREs to CVEs and CCEs.
DR2	**CRE Data Exchange Format (CRE-DEF)** - Standard definition of an exchange format for basic remediation information	• This will be an appendix to the CRE Standard.
DR3	**Extended Remediation Information (ERI)** - Standard definition of desired additional information about a remediation, including mapping to applicable platforms, related vulnerabilities, or configuration issues	• Standard should include a way of mapping CREs to ERIs.
DR4	**ERI Data Exchange Format (ERI-DEF)** - Standard definition of an expression language for the additional information about remediation identified in DR3	• This will be an appendix to the ERI Standard.
DR5	**Remediation Policy Language (RPL)** - Standard way of specifying which policies apply to which classes of assets (implemented as an XML schema)	• Standard should include a way of mapping a policy to an IT asset type. • Standard should include a way of uniquely defining asset types. • Standard should include a way of uniquely defining policy types (e.g., policies for registry keys, file permissions, etc.). • Standard should define level(s) of readability for policy (e.g., by humans, by machine only). • Standard should include a way of defining dates/times in remediation policy and what dates/times (e.g., creation date, implementation date, and expiration date) are required/desired. Standard should be able to handle elapsed time such as 30 days without specifying an actual end date. • Standard should include criteria to be used to select between multiple remediation options. • Standard should define how long assets may defer implementation of a remediation. • Standard should include info on who issued the policy, whom or what it applies to, if it is mandatory or optional, what the policy issuer's authority or scope is and—if multiple options exist—the order of preference for options. • Standard should include a way of stating who can send out policies, who can edit policies so that the Remediation Manager knows from whom to accept policies, and if they may be edited locally. [Note: Need to consider the fact that the standard has no proof of authority; the RM implementation will need to be designed to allow varying login authority, but the standard does not specify how this will be accomplished.]

DR6	**Remediation Tasking Language** (**RTL**) - Standard way of applying specific remediation tasks to specific assets in an enterprise environment (Implemented as an XML schema) (Formerly called Remediation Control Language)	• Standard should include a way of mapping particular remediation tasks to IT asset type and/or Remediation Tool. • Standard should include a way of uniquely defining assets, Remediation Managers, and Remediation Tools. (It should match the Remediation Policy Language standard, DR5.) (RM and RT must have a standard way to use IP address, MAC address, etc., to uniquely identify assets.) • Standard should express what remediation actions, with what values, will be performed on what assets, via what RTs, and by what due date/time. • Standard should define whether remediation tasks are required, allowed, preferred, or prohibited. • Standard should include a way to express the order in which remediation tasks should be performed. • Standard should require human readable information on what policies caused the generation of a remediation task. • Standard should include a unique ID for each remediation task statement that is sent to an RT. • Standards should include a set of tags that identify the manager sending the task.
DR7	**Remediation Tasking Results Language** (**RRF**) - Standard way of reporting the results of an attempted remediation task (Uses a defined XML schema and is part of the RM-to-RT interface control document [ICD])	• Standard should define a way for Remediation Tools and assets to report back to the Remediation Manager what they did and did not do and why. • Standard should define a way of reporting exceptions to policy (POA&Ms) and to remediation tasks. • Standard should include a unique definition of error types (i.e., unsuccessful remediation tasks).
	Common Configuration Enumeration (CCE)	
	Common Vulnerabilities and Exposures (CVE)	
	Scan Results Language - Standard way of reporting scan results; **ARF** (Assessment Results Format) and **ARCAT** (Assessment Results Consumer & Analysis Tool)	• Includes DoD ARF XML from ARCAT, XCCDF, ASR, and OVAL. **Note**: ARCAT is a tool that demonstrates the use of the ARF data exchange standard. • The RM must have a trusted connection to the Scan Results Provider. • The RM must have knowledge of the benchmark used in the scan.
	Active Directory API - Standard interface for tasking and reporting to and from Active Directory Services	
	POA&M Format - Standard for Plans of Actions and Milestones (POA&M) format and content consistent with Netops Data Standards	• Standard is the DoD Format standard. • Consider offering this capability as a plug-in to HBSS with OCIL built on to export reports from POA&M and ask C&A questions.
	Host Information Language - Standard way of reporting host information to the Remediation Manager and Remediation Tool	• This is to be defined.
	ASR Report Format - Standard way for Remediation Managers to log what tasks they sent out, to what Remediation Tools/assets, on what authority, and based on what policy; current status of tasks;	

	and what was reported back	
	Remediation Results Report Format - Standard way for Remediation Tools to log what tasks they received, from whom they received them, and what they did as a result	
	Common Vulnerability Scoring System (CVSS) - Standard way of scoring tasks for task prioritization and risk mitigation activities	
	Help Desk Ticket Format - Standard way to transmit help desk tickets	• This should be based on the Remedy Ticketing System.
	Common Configuration Scoring System (CCSS) - Standard way of scoring and prioritizing configuration compliance tasks	
	Open Checklist Interactive Language (OCIL) - Framework for performing manual checks	• Consider including POA&M formatting.

3.3 Remediation Manager Requirements

RM Requirements, initially developed in 2010, are based on the user scenarios. The tables in this section contain the Remediation Manager requirements along with the increment in which they are expected to be implemented. Reference Implementation Increment 1 was delivered in September 2010, and Increment 2 was delivered in December 2010. Increment 3 had multiple deliveries in March, June, and August 2011. Increment 4 is to be delivered on September 30, 2011.

Table 7: Standards and External Interface Control Documents (ICDs)

Reqmt ID	Requirement	Reference	Increment in which reqmt will be implemented
RM 1.1	The RM shall operate independently of any remediation actions and network remediation tools.		Operational Implementation
RM 1.2	The RM shall be compliant with the standards listed in the previous table.		Operational Implementation
RM 1.3	The RM shall be compliant with the following external ICDs: • Scan Results ICD • Remediation Policy Repository ICD • RT to RM ICD • Host/Asset info ICD • CRE/ERI ICD • Help Desk Ticket ICD		Operational Implementation

Table 8: Remediation Manager Input Requirements

Reqmt ID	Requirement	Reference	Increment in which reqmt will be implemented
RM 2.1	The RM shall accept authenticated scan results consistent with the Scan Results Language Standard and store this data.	User Scenario 1	The RM Reference Implementation Increment 2 meets ARF v0.41. Operational Implementation may implement other scan results.
RM 2.2	The RM shall import policy instructions (e.g., DISA STIGs) consistent with standards requirement DR5 and store this data.	User Scenario 1	RM Reference Implementation Increment 3
RM 2.3	The RM shall import host information consistent with the Host Information Language Standard and store this data.	User Scenario 1	Operational Implementation
RM 2.4	The RM shall import CRE and ERI data consistent with standards requirements DR1, DR2, DR3, and DR4 and store this data.	User Scenario 1	RM Reference Implementation Increment 2 (CRE only) Operational Implementation
RM 2.5	The RM shall accept remediation task results consistent with the Remediation Tasking Results Language standard requirement DR7 and store this data.	User Scenario 2	RM Reference Implementation Increment 1
RM 2.6	The RM database schema shall be compatible with the ARCAT database schema.		RM Reference Implementation Increment 4 (TBR)
RM 2.7	The RM shall accept policy changes made locally from an RT and use these as a local policy for a particular policy group. The exchange format for the policy changes shall be consistent with the RM-to-RT ICD.		Operational Implementation
RM 2.8	The RM shall import other SCAP data, such as CCSS and OCIL information, as needed and store this data.	User Scenario 1	Operational Implementation

Table 9: Remediation Manager Output Requirements

Reqmt ID	Requirement	Reference	Increment in which reqmt will be implemented
RM 3.1	The RM shall output a directive to apply a remediation task consistent with RTL standard requirement DR6.	User Scenario 2	RM Reference Implementation Increment 2
RM 3.2	The RM shall publish Remediation Status reports on all known hosts in its purview consistent with the report standards. These reports may include the status of remediation tasks with their due time, the success or failure of remediation tasks, and explanations for failed remediation tasks. Reports shall be generated which show trends, summaries, remediation counts and other system health measures. (Note: Initially, these reports may be sent through email of a text document. Later reports may be generated through a commercial application such as Crystal Reports.)	User Scenario 3	Operational Implementation

RM 3.3	The RM shall output its event logs, policy hierarchy, and health of RM components in report format consistent with the ASR standards requirement (TBR).	User Scenario 3	Operational Implementation
RM 3.4	The RM shall output help desk tickets consistent with the Help Desk Ticket Format standards requirement.	User Scenario 2	RM Reference Implementation Increment 2
RM 3.5	The RM shall publish POA&M messages consistent with the Netops data standards. (Note: DoD format. Initially, this may be done through email of a text document. May use OCIL.)	User Scenario 1	Operational Implementation
RM 3.6	The RM shall output remediation directives to multiple RTs, each of which may manage one or more hosts/assets.	User Scenario 2	Operational Implementation
RM 3.7	The RM shall do a broadcast of an RTL statement to all RTs. (TBR)		Operational Implementation

Table 10: Remediation Manager User Interface Requirements

Reqmt ID	Requirement	Reference	Increment in which reqmt will be implemented
RM 4.1	The RM shall be managed via a graphical user interface.	User Scenarios 1-7	RM Reference Implementation Increment 3
RM 4.2	The RM shall allow users to choose which remediation action to apply when multiple options are included in policy.	User Scenario 2	Operational Implementation
RM 4.3	The RM shall allow a user to tailor a policy for a given set of assets via a Policy Manager Page based on imported STIG, CRE, and OCIL information.	User Scenario 1	Operational Implementation
RM 4.4	The RM shall allow users to create POA&Ms for policy deviations using the DoD POA&M format.	User Scenario 1	Operational Implementation
RM 4.5	The RM shall monitor and display the health of all the individual components of the RM. This includes components' activities and error events. A user shall be able to view and print any log.	User Scenario 3	Operational Implementation
RM 4.6	The RM shall allow users to edit policies, CREs, ERIs, findings, host information, remediation tasks, remediation results, and reports (TBR). (Need to resolve what a user is allowed to edit.)		Operational Implementation
RM 4.7	The RM shall allow users to assign policies to policy groups. The RM shall allow users to assign hosts directly to a Policy Group or to a Host Group. Host Groups may contain many hosts but shall only be assigned to one Policy Group, as a host shall only belong to one policy group.	User Scenario 1	Operational Implementation
RM 4.8	The RM shall allow a user to generate, view, and print any report. (Note: Need more details for this requirement.)	User Scenario 3	Operational Implementation
RM 4.9	The RM shall verify and report to a user that it has successfully or unsuccessfully integrated with ARCAT.		Operational Implementation
RM 4.10	The RM shall allow the user to make modifications to a remediation task that has not yet been sent to an RT, for example, allowing a user to override a task due date/time. All changes shall be recorded in a log. (Note: Currently, we only want to allow a user to modify the due time of a task or to cancel a task. User may not make any other modifications to a		Operational Implementation

	task. Also, user may not make any modifications to a task that has already been sent to an RT.)		
RM 4.11	The RM shall display critical status information on its home page. This information may include remediation tasks that have not been completed and are past their due dates/times, new policies that have come in, hosts that need to be assigned to a policy group, and hosts that have not been successfully remediated.		Operational Implementation
RM 4.12	The RM shall provide pages that show what policies are in a particular policy group and what hosts are in a particular policy group.	User Scenario 2	Operational Implementation
RM 4.13	The RM shall alert the user about newly discovered hosts and allow the user to assign them to a Policy Group either directly or through a Host Group assignment.	User Scenario 1	Operational Implementation
RM 4.14	The RM shall alert the user when a help desk ticket is created.	User Scenario 2	Operational Implementation
RM 4.15	The RM shall provide a Task Status Page that shows all tasks generated by the RM and their status. Tasks shall be color coded based on their status. The user shall be able to click on a specific task to bring up a page with all the information about that particular task.	User Scenario 2	Operational Implementation
RM 4.16	The RM shall allow a user to create a help desk ticket.	User Scenario 2	Operational Implementation
RM 4.17	The RM shall provide a Host Management Page that shows the status of all known hosts in the RM's purview. Hosts shall be colored red if they have failed any remediation task. Hosts shall be colored yellow if remediation tasks that have been sent have not been completed. Hosts shall be colored green if they are in compliance with their local policy and have no outstanding tasks or findings. Hosts shall be colored gray if there have been no scan results received for a defined period of time. The user shall be able to click on a host to bring up a page with all the information about that host and each task sent to it.	User Scenario 2	Operational Implementation
4.18	The RM shall provide displays of policy hierarchy, event logs, component health, and help desk tickets.		Operational Implementation
4.19	The RM shall allow the user to generate a special task that shall immediately be sent to the appropriate hosts.	User Scenario 4	Operational Implementation
4.20	The RM shall allow the user to reassign a policy to a different Policy Group based on as yet to be defined attributes and restrictions.	User Scenario 5	Operational Implementation
4.21	The RM shall allow the user to reassign hosts to a different Host Group or Policy Group based on as yet to be defined attributes and restrictions. The hosts shall be able to be moved one at a time or in bundles by selecting multiple hosts on the Host Management Page.	User Scenario 5	Operational Implementation
4.22	The RM shall allow the user to reassign a Host Group to a different Policy Group based on as yet to be defined attributes and restrictions.	User Scenario 5	Operational Implementation
4.23	The RM shall allow the user to create a Test PG.	User Scenario 6	Operational Implementation
4.24	The RM shall allow an administrator to create logins for a variety of user roles, configure options, establish Policy Groups and Host Groups, and store this information.	User Scenario 7	Operational Implementation

Table 11: Remediation Manager Internal Interface Requirements

Reqmt ID	Requirement	Reference	Increment in which reqmt will be implemented
RM 5.1	The RM shall examine a host's scan results and the local policy attached to a host's assigned Policy Group for findings against the local policy. If found, the RM shall map the finding to its corresponding CRE(s) and generate the appropriate remediation task(s) with due times.	User Scenario 2	RM Reference Implementation Increment 2
RM 5.2	The RM shall determine the appropriate Remediation Tool for each remediation task that has been generated.	User Scenario 2	RM Reference Implementation Increment 2
RM 5.3	If the RM does not receive a response from the Remediation Tool and the task due date/time has passed, then the RM shall mark the task result as "failed" and shall generate a help desk ticket. (Need to consider scenario in which RT is turned off or otherwise unavailable.)	User Scenario 2	Operational Implementation
RM 5.4	The RM shall provide the ability for a user to create a Test Group and verify through that Test Group that a remediation task performs as expected before applying it to a Policy Group.	User Scenario 6	Operational Implementation
RM 5.5	The RM shall assign the following status to a remediation task: • "new" for tasks that have been generated but not yet sent to a Remediation Tool • "in process" for tasks that have been sent to a Remediation Tool but no response has been received from the RT • "completed" for tasks that the RT has responded to say it completed • "not completed" for tasks that the RT has responded that it has not yet completed but are not past their due date/time. • "failed" for tasks for which the RT has not sent any response and the due date/time has expired. • "reject" for tasks which the RT has responded that it will not complete. • "expired" for tasks which the RT has responded to but has not yet completed and the expected due date/time has passed.	User Scenario 2	RM Reference Implementation Increment 3 implemented a portion of these assignments
RM 5.6	If the CVSS for a remediation task/finding is less than or equal to a specified value, then the RM shall status that remediation task/finding as "completed." (TBR)		Operational Implementation
RM 5.7	The RM shall be able to verify whether or not it has successfully integrated with ARCAT.		Operational Implementation
RM 5.8	The RM shall be able to automatically assign hosts to policy groups.		Operational Implementation
RM 5.9	The RM shall be able to send email notifications to users when task status has changed.		Operational Implementation
RM 5.10	If the RM receives a new scan result for a task that has been statused "completed" that indicates it was not completed, then the RM shall generate a help desk ticket.	User Scenario 2	RM Reference Implementation Increment 3

RM 5.11	When a user creates a POA&M for a local policy that is assigned to a Policy Group, the RM shall map this POA&M to the local policy and store it.	User Scenario 1	Operational Implementation
RM 5.12	When the RM receives scan results on an unknown host, the RM shall place the unknown host into the unassigned Policy Group and send an alert to the user that the host needs to be assigned to a Policy Group along with any policy compliance information that has been received. The RM shall status this host with the color red until the user assigns it to a Policy Group.	User Scenario 1	Operational Implementation
RM 5.13	When the RM receives a repeat scan finding for a host that was previously sent a remediation task to correct the finding and that remediation task due time has passed, then the RM shall mark that task status as "failed," create a help desk ticket, and send an alert to the user.	User Scenario 2	Operational Implementation (for alert to the user)
RM 5.14	When the RM receives a repeat scan finding for a host and that finding does not match the local policy for the host's Policy Group, the RM shall generate a help desk ticket and send an alert to the user.	User Scenario 2	Operational Implementation
RM 5.15	The RM shall ensure that a specific CRE is only sent once to a specific host.	User Scenario 2	RM Reference Implementation Increment 3
RM 5.16	When the RM discovers a noncompliant finding against a local policy item that has more than one potential remediation that could be performed, the RM shall generate a help desk ticket and send an alert to the user. (The Operational Implementation may allow the user to select a remediation or generate a help desk ticket.)	User Scenario 2	Operational Implementation
RM 5.17	If the RM has received a response from the RT regarding a task but the RT has not completed it and the task due date/time has passed, then the RM shall mark the task result as "expired" and shall generate a help desk ticket.	User Scenario 2	Operational Implementation
RM 5.18	The RM shall be capable of sending out a special remediation task that is not based on scan results.	User Scenario 4	Operational Implementation

Table 12: Remediation Manager Non-Functional (Quality Attribute) and Miscellaneous Requirements

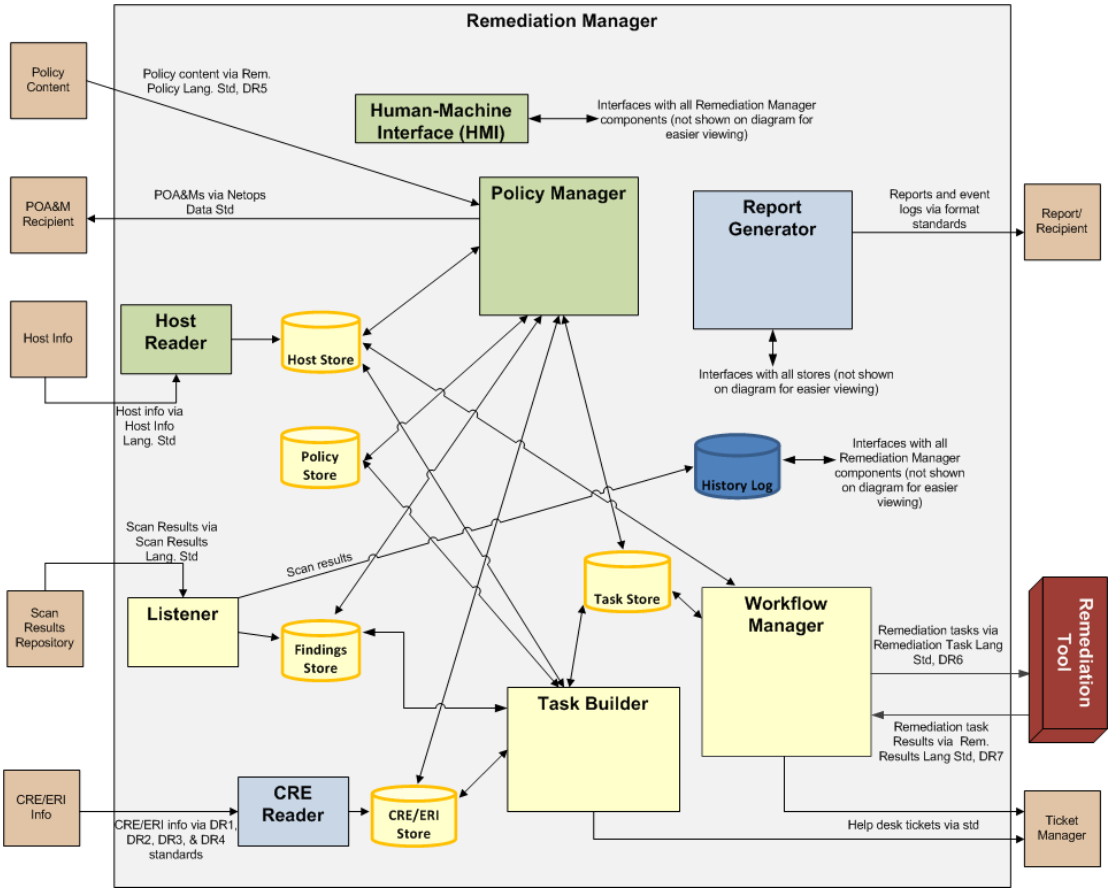
Reqmt ID	Requirement	Reference	Increment in which reqmt will be implemented
RM 6.1	The RM shall be scalable and configurable for local as well as centralized management.		Operational Implementation
RM 6.2	The RM shall be capable of communicating using secure methods.		Operational Implementation
RM 6.3	The RM shall provide secure identification and authentication mechanisms between all external components.		Operational Implementation
RM 6.4	The RM shall support DoD PKI certificates.		Operational Implementation
RM 6.5	The RM shall not interfere with the operation of DoD-mandated Information assurance (IA) tools (e.g., anti-virus programs).		Operational Implementation
RM 6.6	The RM shall be able to interface with third-party network operations tools such as reporting and security information management systems.		Operational Implementation

RM 6.7	The RM shall ensure the integrity of stored data.		Operational Implementation
RM 6.8	The RM shall provide fail-over and/or redundancy capabilities.		Operational Implementation
RM 6.9	The RM shall be able to run in a virtual environment.		RM Reference Implementation Increment 2
RM 6.10	The RM shall provide integrity controls to protect against compromise of the remediation solution.		Operational Implementation
RM 6.11	The RM shall be manageable by local administrators who shall be able to configure role-based access controls and permission groups. (Note: Need more security requirements.)	User Scenario 7	Operational Implementation

4 Current Reference Implementation Architecture

Figure 4 illustrates the top-level components of the remediation manager architecture. These components were developed in four increments, two in 2010 and two in 2011:

- Increment 1, delivered in September 2010, includes the Workflow Manager, associated data stores, and the Remediation Manager interface to the Remediation Tool.
- Increment 2, delivered in December 2010, includes Increment 1 plus the Task Builder, Listener, and associated data stores and interfaces.
- Increment 3, delivered in three sub-increments (March, June, and August, 2011) includes updates to Increment 1 and 2 plus policy management functions (policy assignment and tailoring), a graphical user interface, and a tasking/results interface with a remediation tool.
- Increment 4, delivered in September 2011, is an enhancement to Increment 3 that employs the ARCAT[8] database schema for relevant remediation data store items.



Note: Yellow-shaded components were initially implemented in 2010 and updated in 2011; green-shaded components were initially developed in 2011.

Figure 4: Remediation Manager Conceptual Architecture

[8] ARCAT stands for Assessment Results Consumer & Analysis Tool, a DoD reference implementation that facilitates use of DoD Assessment Results Format (ARF) 0.41 documents.

4.1 Traceability of Architecture Components to Capabilities and Requirements

Table 13 illustrates traceability of the Remediation Manager Functions developed in 2010 and 2011 to the top-level capabilities described in Section 2.6, the system-level requirements listed in Table 3, and the architectural components identified in Figure 4. The functional requirements for each architecture component shown in the figure are documented in Section 3.3. Note that each of the user scenarios documented in Section 3 requires the interaction of several remediation manager functions.

Table 13: Traceability of Remediation Manager Functions to Top-Level Function, System-Level Requirement, and Architecture Component

Category	Remediation Manager Functions *gray text: function not yet developed* *red text: emerging remediation standards exercised; STD*		Top-Level Function* (Section 2.6)	System-Level Req't (Table 3)	Architecture Component (Figure 4)	Implemented	
						2010	2011
User Interface	Control Remediation Manager Functions		Derived*	Derived*	HMI		X
	Display Host Scan Findings		Derived	Derived	HMI		X
	Manage Remediation Policy		1	7, 4	PM		X
	Develop POA&Ms (local policy tailoring justification)		1	8	PM		
	Manage Host Information		1	Derived	HR, PM		X
	Display Analysis Reports		4	Derived	RG		
	Display Remediation Tasking Status		Derived	Derived	RG		X
External Input	Ingest Policy (from Remediation Authority, RA)	RPL	1	2	PM		
	Ingest Remediation Definitions (from RA)	CRE and ERI	1	2	CR		
	Ingest Findings (from host scans)	DOD ARF 0.41	2	1	LIS	X	X
	Ingest Remediation Tasking Status (from RT)	RRF	3	10	WFM	X	X
Database[9]	Store Scan Findings	DoD ARF 0.41	Derived	Derived	FS		X
	Store Remediation Policy	RPL	Derived	Derived	PS	X	X
	Store Remediation Definitions	CRE and ERI	Derived	Derived	CES	X	X
	Store Host Information	DoD ARF 0.41	Derived	Derived	HS		
Internal Processing	Tailor Remediation Policy		1	7	PM		X
	Build Remediation Tasks		2	5, 6	TB	X	X
	Analyze Remediation Tasking Status		4	Derived	RG		
External Output	Transmit Remediation Tasking (to Remediation RT)	RTL	3	3	WFM	X	X
	Transmit Remediation Tasking Status Reports (to RA)		4	9, 11	RG		

*Derived functions and requirements are those needed to support user-specified functions and requirements.

[9] The Remediation Manager database schema was designed to be consistent with the DoD ARCAT data schema wherever possible (i.e., RM and ARCAT data items that are identical share the same structure).

4.2 Relationship to Security Automation Standards

As shown in Table 13, security automation standards are key building blocks for reference implementation input, output, and database functions. In particular, development and testing of the remediation-specific automation standards—CRE, ERI, RTL, RRF, and RPL—are the primary reason for developing the reference implementation. Requirements for these standards are documented in Table 6.

In the 2010-2011 time frame, we developed versions of CRE, RTL, and RRF for use by the Remediation Manager and Remediation Tool. Draft specifications for CRE and ERI (ERI supplements the information provided by CRE) are under development by our colleagues at MITRE, who plan to release them for comment in the coming months. We expect to see specifications for RTL and RRF next. We did not develop a test version of RPL for the 2010-2011 reference implementation effort. For RPL development to advance, coordination with an existing remediation authority is desirable.

Existing standards for security content automation also play a central role in remediation. These standards belong to the suite of standards in the Security Content Protocol [NIST 2011]. Examples are standards describing CCE [MITRE 2011b] and CVE [MITRE 2011a].

5 Current Reference Implementation Capabilities

The 2011 Remediation Manager is a web-based application that includes four main pages: Home, Assessment Results, Policy Manager, and Task Status. This section includes screenshots for these pages along with a description of the capabilities provided. A fifth page, a Host Manager page, was under consideration for the reference implementation but has not been developed. Capabilities to be provided by the Host Manager page are described in Section 5.5.

5.1 Remediation Manager Home Page

The remediation manager home screen, shown in Figure 5, has two main capabilities: A **Statistics** display and a **Control Panel.**

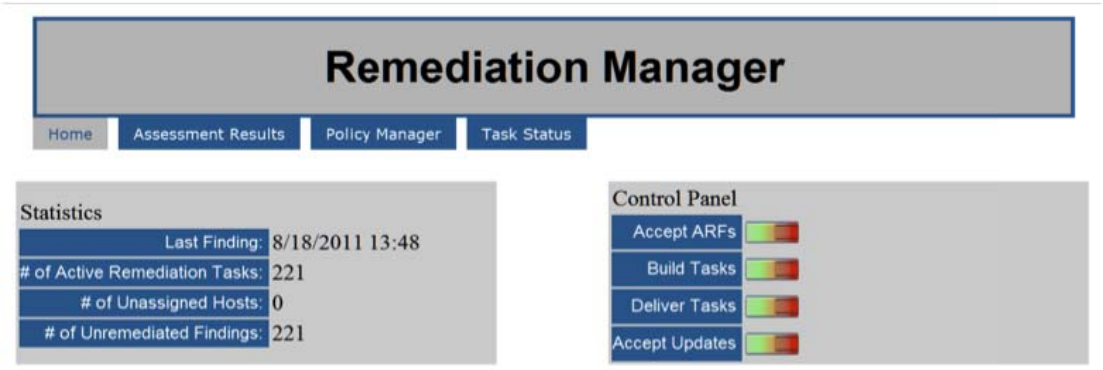


Figure 5: Remediation Manager Home Page Screen Shot

In the 2011 implementation, the **Statistics** display includes the following items:

- **Last Finding**: Date and time of the last finding the remediation manager received from a scanning device
- **# of Active Remediation Tasks**: Number of tasks the remediation manager has generated (or generated and sent to a remediation tool) for which status has not yet been received
- **# of Unassigned Hosts**: Number of hosts not assigned to a Policy Group
- **# of Unremediated Findings**: Number of findings for which tasks have not yet been generated plus **# of Active Remediation Tasks**

  Note: An operational remediation manager may include different or additional items on the statistics display, for example, number of repeat findings for a host.

The **Control Panel** consists of "toggle switches" that control the following four functions:

- **Accept ARFs**: Turn on/off the function that ingests ARFs from scanners.
- **Build Tasks**: Turn on/off the function that maps findings in ARFs (CCEs and CVEs) to remediations (CREs) per DoD and/or local remediation policy (eventually to be expressed in RPL) and builds remediation tasks (in RTL).

- **Deliver Tasks**: Turn on/off the function to deposit remediation tasks (the RTL files created in **Build Tasks**) into the task store for a Remediation Tool to pick up.

  Note: In an operational remediation manager, task delivery may be implemented using a different communication mechanism.

- **Accept Updates**: Turn on/off the function to read status updates (in RRF) deposited to the task store by a Remediation Tool and update the Task Status accordingly.

  Note: An operational remediation manager may include different or additional items on the control panel. For example, it may add a switch to turn on/off notifications and it may not include separate switches for task building and delivery

5.2 Remediation Manager Assessment Results Page

The Assessment Results page, illustrated in Figure 6, enables the user to review the list of compliance issues for each host. The CCE related to the issue is displayed along with a description of the issue and its remediation status. Future capabilities under consideration include the ability to search for a particular host or find a host using a pull-down window, and perhaps to show the number of hosts having a particular compliance issue.

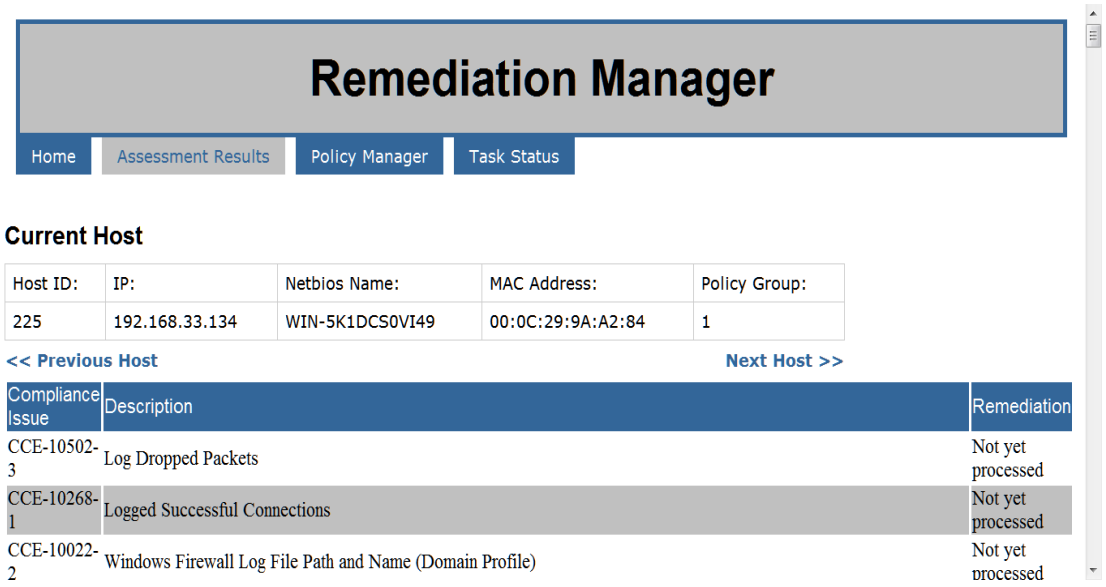


Figure 6: Remediation Manager Assessment Results Page Screenshot

5.3 Remediation Manager Policy Manager Pages

Policy Manager pages enable the user to edit, override, or mitigate DoD policy. The DoD policy for a given operating system or platform is the default policy and is assigned to the top-level policy group (PG). Lower-level PGs inherit policy from the top-level PG. The user can tailor (edit) policy for these lower-level PGs.

Figure 7 illustrates the view of the Policy Manager page the user first sees when clicking on the Policy Manager tab. This view shows the remediation policy, which consists of a mapping from

compliance issues (i.e., CCE) to remediation handling details (i.e., CRE). To examine or edit policy for a different PG, the user would click the Next Policy Group >> or << Previous Policy Group link. An operational remediation manager would provide the capability to search or browse for a particular PG.

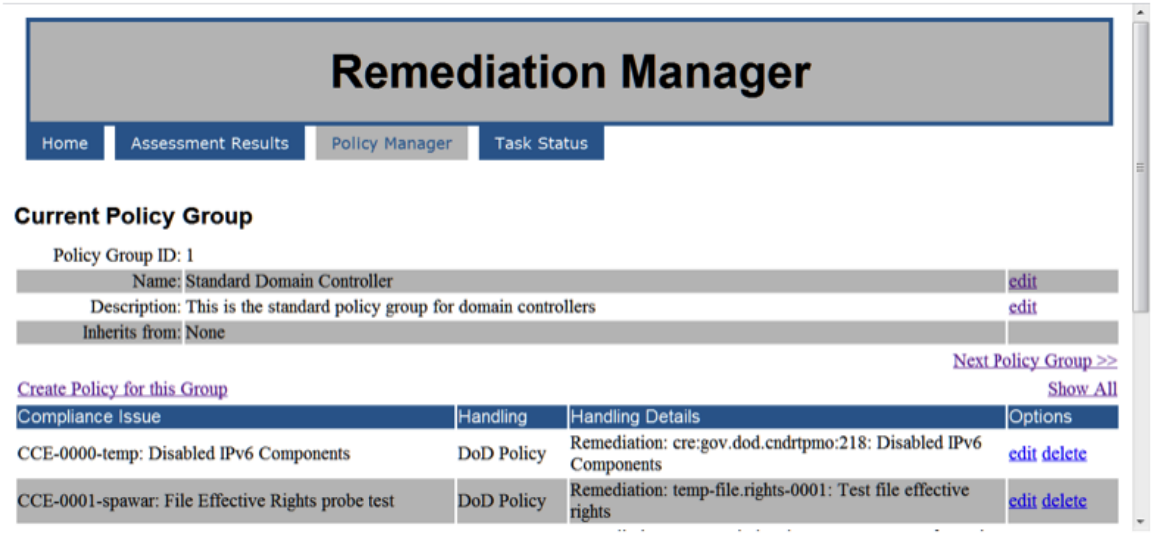


Figure 7: Remediation Manager Policy Manager Page Screenshot

The user can edit the mapping of CCEs to CREs by clicking the "edit" link in the "options" column. When the user clicks "edit," the compliance issue (CCE) and handling details (CRE) are replaced with drop-down boxes as shown in Figure 8.

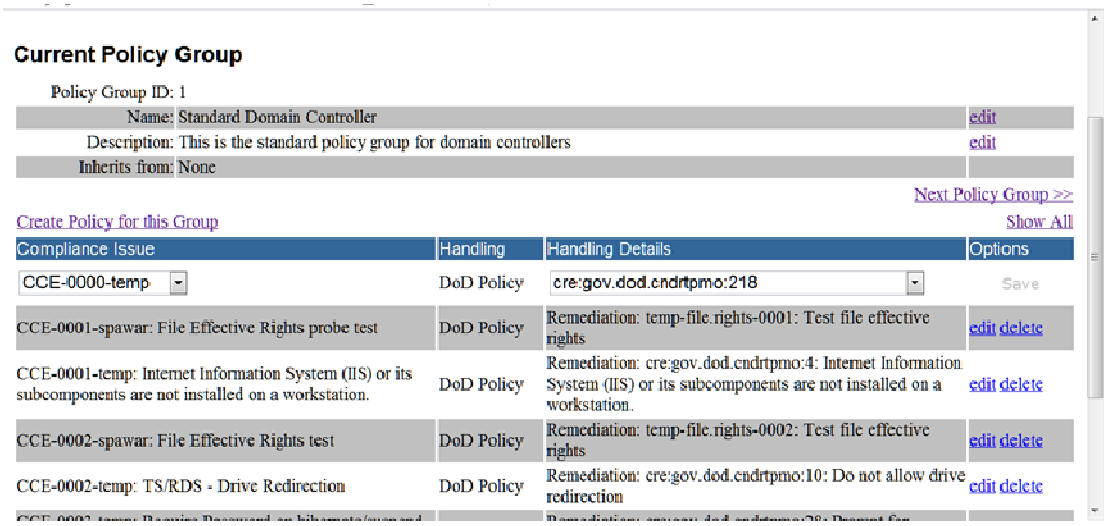


Figure 8: Remediation Manager Policy Manager Page Edit Screenshot

The user may then change the mapping from compliance issue (CCE) to handling details (CRE), clicking "Save" to return to the main page view.

Lower-level policies (i.e., policies tailored from the default policy) will include options to "override" or "mitigate" policy as shown in Figure 9.

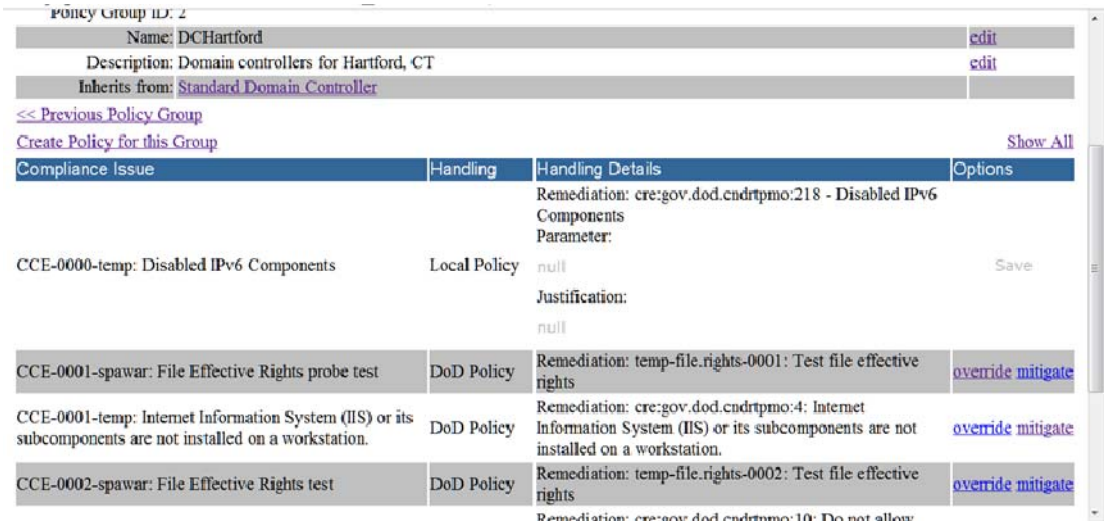


Figure 9: Remediation Manager Policy Manager Page Override Screenshot

When the user clicks "override," the "Handling Details" column enables editing of a parameter and entry of a justification for overriding DoD policy for that parameter. The user clicks "Save" to exit the override mode. Mitigation works in a similar way, except it does not enable changes to handling details. Instead, it allows the user to describe the mitigation to be applied in lieu of remediation and to provide a justification, as shown in Figure 10. An operational remediation manager would provide the capability to document a POA&M for both policy overrides and mitigations.

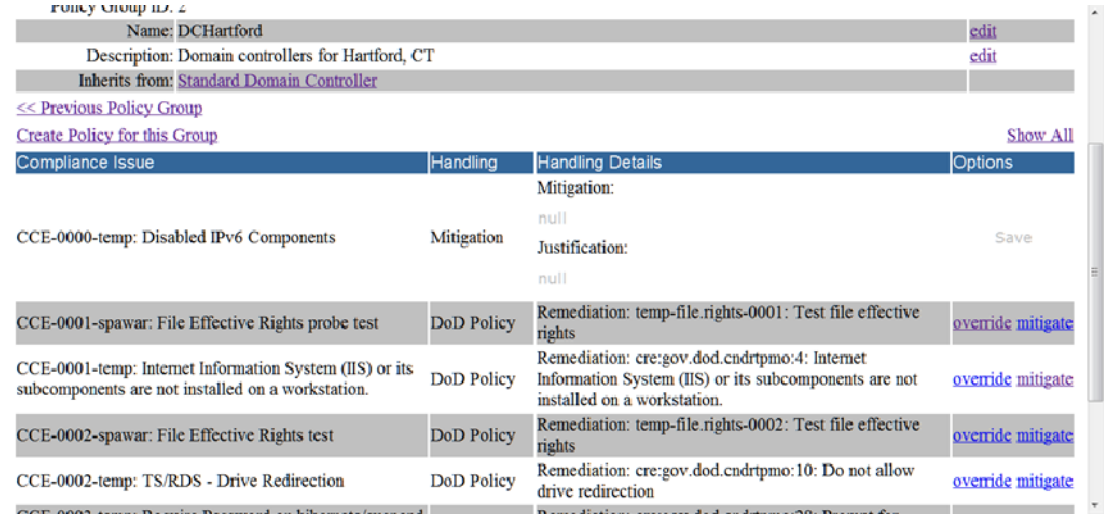
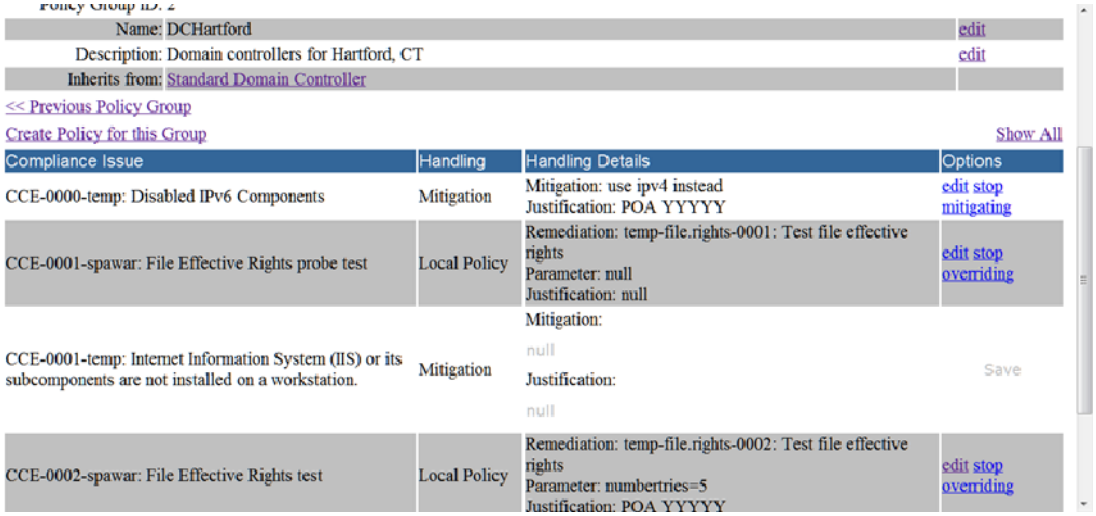


Figure 10: Remediation Manager Policy Manager Page Mitigate Screenshot

After the user has saved the override or mitigation information, the Options column provides choices to edit and to either stop mitigating or stop overriding (as applicable), as shown in Figure 11.



Policy Group ID: 2

Name:	DCHartford	edit
Description:	Domain controllers for Hartford, CT	edit
Inherits from:	Standard Domain Controller	

<< Previous Policy Group

Create Policy for this Group — Show All

Compliance Issue	Handling	Handling Details	Options
CCE-0000-temp: Disabled IPv6 Components	Mitigation	Mitigation: use ipv4 instead Justification: POA YYYYY	edit stop mitigating
CCE-0001-spawar: File Effective Rights probe test	Local Policy	Remediation: temp-file.rights-0001: Test file effective rights Parameter: null Justification: null	edit stop overriding
CCE-0001-temp: Internet Information System (IIS) or its subcomponents are not installed on a workstation.	Mitigation	Mitigation: null Justification: null	Save
CCE-0002-spawar: File Effective Rights test	Local Policy	Remediation: temp-file.rights-0002: Test file effective rights Parameter: numbertries=5 Justification: POA YYYYY	edit stop overriding

Figure 11: Remediation Manager Policy Manager Page Screenshot (after overriding/mitigating)

5.4 Remediation Manager Task Status Page

In the 2011 version of the Remediation Manager, the Task Status page, as illustrated in Figure 12, provides a simple display of the status of each remediation task as follows:

- NEW: Remediation Manager has generated the task but has not yet sent it.
- INPROCESS: Remediation Manager has sent the task to a Remediation Tool, no response has yet been received, and the task is not past its due date.
- COMPLETED: Remediation Manager has sent the task to a Remediation Tool and received a "successful completion" status from the tool.
- FAILED: Remediation Manager has sent the task to a Remediation Tool and has received an "unsuccessful completion" status from the tool.

Figure 12: Remediation Manager Task Status Page Screenshot

User Scenario 2 in Section 3.1.2 refined and expanded these options as shown in Table 4. This refinement should be considered for an operational remediation manager.

5.4.1 Task Status Page: Changes Under Consideration

In addition to the refinement for remediation task status values, we anticipate additional changes to the status display capability for an Operational Implementation. First, the **Task Status** page would become a **Host Task Status Summary Page** that would show, for example, the following for each host for which tasks had been generated:

- Host Name
- Summary Status Value for the host, color coded as follows (this list uses the refined status values from Table 4):
  - Green: All tasks for the host have been reported as COMPLETED.
  - Yellow: All tasks for the host have been reported as either NEW, INPROCESS, or COMPLETED; no tasks have been reported as FAILED/EXPIRED or REJECTED.
  - Red: One or more remediation tasks have been reported as FAILED/EXPIRED or REJECTED.

Other desirable features for the **Host Task Status Summary** page include the ability to search for a host, to sort on status value, and to remove findings older than a certain date from the display (removed findings would be retained in the history store and could be redisplayed on request).

From the **Host Task Status Summary** page, the user would be able to click on any host and be taken to the **Host Manager Page**. There, the user could drill down to a **Host Task Status Details** page for that host. This page would identify the status for each specific task that has been completed, is in process, or has not been completed and the details (i.e., the remediation) for each task.

5.5 Host Manager Page (not currently implemented)

Near the end of our 2011 effort, we discussed requirements for a new Host Manager page (see User Scenarios 2 and 3, Figure 3, and the requirements in Section 3.3). This page would provide the following capabilities:

- assign one or more hosts a Host Group
- move one or more hosts to a different Host Group
- assign one or more hosts (or Host Groups) to a Policy Group
- move one or more hosts (or Host Groups) to a different Policy Group
- display detailed host task status (as described in Section 5.4.1)

6 Observations, Next Steps, and Conclusions

6.1 Observations and Questions for Consideration

In developing the remediation manager, we identified several questions for consideration based on our work and our participation in discussions and forums on security automation. Some of these questions were identified in our 2010 report, although we have added a few based on our observations this year. These questions involve concepts of operations for end-to-end DoD remediation, both the ideal and what can be achieved in the near term. Key topics the team has identified as needing further exploration, discussion, experimentation, and articulation include

- the balance between automation and user intervention in the Remediation Manager and the Remediation Tool as well as the allocation of functions between these two elements of the end-to-end remediation solution
- hierarchical and peer-to-peer relationships with respect to reporting and other types of information sharing
- the extent to which standards-based remediation management can be centralized and coordinated across DoD, and different architectural strategies for accomplishing key coordination goals
- policy management, including automating the evaluation of new and updated policies to identify conflicts and compromises, keeping policies current and consistent, and adjudicating and reporting conflicts between global and local policies
- remediation of resources in addition to end-point systems such as network devices
- remediation in the context of cloud computing and desktop virtualization

These and other topics need to be shared and discussed within the security automation standards community, DoD Configuration Management community, and security solutions vendor community.

6.2 Candidate Next Steps

At the conclusion of our two-year effort, we foresee future work for remediation automation in a number of areas.

6.2.1 Developing and Refining User Scenarios, Standards, and Remediation Manager Requirements: User and Vendor Engagement

As previously explained, not all the detailed requirements identified in Section 3 have been implemented or even reviewed in detail. Also, not all requirements have been discovered. Discussions of both remediation management capabilities and automation standards, such as those fostered by Security Automation Developer Days and the Information Technology Security Automation Conference (ITSAC), should intensify to explore key areas in greater depth.

It should be noted that the emphasis for the remediation manager reference implementation was strictly on functional requirements. For operational use, requirements for non-functional (e.g.,

quality attribute and performance) features must also be analyzed and built into the implementation.

6.2.2 Testing Evolving Standards and Exploring Additional Remediation Management Capabilities

As we further develop the remediation standards, we will need a way to continue testing the them in a demonstration-type environment such as in the SEI Remediation Manager and SPAWAR Remediation Tool. At the same time, we could test key additional capabilities, for example, deadline setting and notification, prioritization, and ticketing; the ability to interface with a remediation authority for policy information; creation of POA&Ms to address tailored policy; storing and processing tool capability data; and results and status logging and reporting.

6.2.3 Evolving the Remediation Vision for the DoD Enterprise: The Impact of Virtualization

With the move from desktop computing resources and data storage to virtual environments, the remediation management concept must expand from a focus on end-point machines to include a variety of resources, mechanisms, and assets.

6.2.4 Extending Remediation Manager Capabilities to Address Complex, Dynamic Situations

Looking toward a future in which standards-based automation is commonplace in remediating vulnerabilities and compliance issues, it is likely that updates to configuration policy will occur more frequently. This may present challenges in ensuring consistency of policy across the remediation management life cycle (i.e., from scan, to remediation, to reporting). The SEI recommends work to develop a smart policy manager to analyze policies when updates or overrides (policy tailoring events) occur and warn of possible inconsistencies, ambiguities, or attacks.

6.2.5 Applying Measurement and Analysis to Support both Enterprise and Local Decision Making

Remediation management is a data-intensive activity performed in a dynamic environment. Statistics logged by the remediation manager could be analyzed to provide indicators related to remediation tasks, policy tailoring, repeat remediations, and other data that might facilitate diagnosing trends or identifying issues with particular assets or asset categories. Data could also be collected and used to determine which remediation methods and tools provide the most (and least) benefit and value.

6.3 Conclusions

Managing the configuration of computing assets on DoD networks is essential to national security. Doing so in a timely fashion requires the application of standards and automation to identify and analyze findings from vulnerability and compliance scans, express and execute remediation policy, and communicate current status. Today's vulnerability and compliance remediation solutions either rely heavily on manual support or employ proprietary expressions of common security data items, complicating or precluding interoperability and increasing remediation time. In dealing with adversaries who operate at internet speed, we cannot afford to

be slow in identifying and applying remediations to thwart attacks. While standards have been developed and applied to identify vulnerabilities and compliance issues, work remains to develop standards to support automated remediation of these issues. Our efforts to advance remediation standards development through the remediation research project are a step in this direction.

Appendix Acronym List

ARCAT	Assessment Results Consumer & Analysis Tool, software system reference implementation to realize, demonstrate, and promote the use of the Assessment Results Format (ARF) and other Data Exchange Standards (DES)[10]
ARF	Assessment Results Format, an XML-based data exchange standard developed from Net D schemas for describing assessment results grouped by device[11]
ASR	Assessment Summary Results, a data exchange standard for describing assessment results grouped by individual findings[12]
CCE	Common Configuration Enumeration (CCE™) [MITRE 2011b]
CCSS	Common Configuration Scoring System
CRE	Common Remediation Enumeration. A CRE entry is a set of actions taken to remediate a vulnerability or misconfiguration on a host. The enumerated list of all standardized CREs is itself referred to as the CRE [Waltermire 2011, p. 5].
CVE	Common Vulnerabilities and Exposures (CVE®) [MITRE 2011a]
DEF	Data Exchange Format
DR	Derived Requirement
ERI	Extended Remediation Information
GPO	Group Policy Object
HG	Host Group
IA	Information Assurance
ICD	Interface Control Document
ITSAC	Information Technology Security Automation Conference
NVD	National Vulnerability Database
OCIL	Open Checklist Interactive Language
OVAL	Open Vulnerability and Assessment Language
PG	Policy Group
POA&M	Plan of Action and Milestones
RPL	Remediation Policy Language
RRF	Remediation Results Format (also known as Remediation Results, Remediation Results Language, and Remediation Tasking Results) [Waltermire 2011, p. 3]
RTL	Remediation Tasking Language (formerly Remediation Control Language) [Waltermire 2011, p. 8]
SCAP	Security Content Automation Protocol
STIG	Security Technical Information Guide
XCCDF	eXtensible Configuration Checklist Description Format, a specification language for writing security checklists, benchmarks, and related kinds of documents

[10] U.S. Department of Defense. *Software Requirements Specification, Assessment Results Consumer & Analysis Tool (ARCAT) Spiral Two.* November 6, 2009.

[11] U.S. Department of Defense. *Assessment Results Format XML Specification, version 0.41*. http://metadata.dod.mil/mdr/ns/netops/shared_data/arf_index_page/0.41 (sponsored access required) 2010.

[12] U.S. Department of Defense. *Assessment Summary Results Format v 0.41 draft.* September 12, 2009.

Bibliography

URLs are valid as of the publication date of this document.

[DHS 2011]
Department of Homeland Security. *Enabling Distributed Security in Cyberspace: Building a Healthy and Resilient Cyber Ecosystem with Automated Collective Action.* 2011.http://www.dhs.gov/files/publications/gc_1302028618408.shtm

[MITRE 2011a]
MITRE. *Common Vulnerabilities and Exposures*. http://cve.mitre.org/ (2011).

[MITRE 2011b]
MITRE. *Common Configuration Enumeration*. http://cce.mitre.org/ (2011).

[NIST 2011] National Institute of Standards and Technology. *The Technical Specification for the Security Content Automation Protocol (SCAP): SCAP Version 1.2 – Draft* (NIST Special Publication 800-126r2). 2011.

[Waltermire 2011]
Waltermire, D., Johnson, C., Kerr, M., Wojcik, M., & Wunder, J. *Proposed Open Specifications for Enterprise Information Security Remediation – Draft* (NIST Interagency Report 7670). NIST, 2011.

Relevant Websites

Abbreviation	Title	URL
ARF	Assessment Results Format (DoD version 0.41)	http://metadata.dod.mil/mdr/ns/netops/shared_data/arf_index_page/0.41
CCE	Common Configuration Enumeration	http://cce.mitre.org/
CPE	Common Platform Enumeration	http://cpe.mitre.org/
CVE	Common Vulnerabilities and Exposures	http://cve.mitre.org/
NIST CSD	National Institute of Standards and Technology Computer Security Division	http://csrc.nist.gov/
NVD	National Vulnerability Database	http://nvd.nist.gov/
SCAP	Security Content Automation Protocol	http://scap.nist.gov/
XCCDF	eXtensible Configuration Checklist Description Format, a specification language for writing security checklists, benchmarks, and related kinds of documents.	http://scap.nist.gov/specifications/xccdf/

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
(Leave Blank)	December 2011	Final

4. TITLE AND SUBTITLE	5. FUNDING NUMBERS
Standards-Based Automated Remediation: A Remediation Manager Reference Implementation, 2011 Update	FA8721-05-C-0003

6. AUTHOR(S)

Sagar Chaki, Rita Creel, Jeff Davenport, Mike Kinney, Benjamin McCormick, Mary Popeck

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)	8. PERFORMING ORGANIZATION REPORT NUMBER
Software Engineering Institute Carnegie Mellon University Pittsburgh, PA 15213	CMU/SEI-2011-SR-016

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSORING/MONITORING AGENCY REPORT NUMBER
ESC/CAA 20 Schilling Circle, Building 1305, 3rd Floor Hanscom AFB, MA 01731-2125	

11. SUPPLEMENTARY NOTES

12A DISTRIBUTION/AVAILABILITY STATEMENT	12B DISTRIBUTION CODE
Unclassified/Unlimited, DTIC, NTIS	

13. ABSTRACT (MAXIMUM 200 WORDS)

This report describes the Software Engineering Institute's (SEI's) 2011 work for the National Security Agency (NSA) to develop standards for automated remediation of vulnerabilities and compliance issues on Department of Defense (DoD) networked systems. The SEI developed a remediation manager reference implementation that demonstrates how evolving standards can communicate and process information on vulnerabilities, compliance issues, remediation policy, and remediation actions. An earlier report, *Standards-Based Automated Remediation: A Remediation Manager Reference Implementation* (CMU/SEI-11-SR-007), described the project's concept, vision, scope, requirements, and the remediation manager implementation as of December 30, 2010. Since then, the SEI has analyzed additional user scenarios, continued remediation standards development, and added new capabilities to the reference implementation.

The remediation manager can employ standards throughout the compliance issue remediation cycle. Using common formats and languages, the reference implementation ingests scan findings, extracts host compliance issues and vulnerabilities, maps them to remediation actions, builds remediation tasks, transmits remediation tasks to a Remediation Tool on a host system, and receives remediation task execution status from the Remediation Tool. In 2011 the SEI added a standards-based remediation policy management capability, enabling users to examine, tailor, and apply standard DoD policy to meet local needs.

14. SUBJECT TERMS	15. NUMBER OF PAGES
automated remediation, computer security, configuration compliance, information assurance, open remediation specification, policy-based remediation, remediation, Remediation Manager, remediation policy, Remediation Tool, Security Content Automation Protocol (SCAP), security noncompliance, security automation standards, standards-based automated remediation, vulnerability	57

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	UL